

Claims

We claim:

1. A medium configured to perform time-bounded execution of a program,
5 wherein the medium is configured to perform:
 - initiating a timed program execution process, wherein the timed program execution process is operable to execute a program in a time-bounded manner;
 - initiating a timeout process;
 - configuring a timeout event;
 - 10 the timed program execution process performing a time-bounded execution of the program, comprising:
 - a) determining and storing a rollback state for the program;
 - b) if the timeout event has not occurred, executing the program, wherein, during said executing, if the timeout event occurs,
 - 15 c) the timeout process setting the timed program execution process to the rollback state, and disabling the timeout event; and
 - d) the timed program execution process resuming executing the program based on the rollback state with a timeout condition;
 - e) performing a program exit procedure;
 - 20 disabling the timeout event;
 - terminating the timeout process; and
 - terminating the timed program execution process.
2. The medium of claim 1, wherein e) further comprises:
25 clearing the rollback state.
3. The medium of claim 2, wherein the medium is further configured to perform:
iteratively performing said time-bounded execution of a plurality of programs.

4. The medium of claim 3, wherein said iteratively performing comprises:
for each of the plurality of programs, performing a) through e).

5 5. The medium of claim 4,
wherein e) further comprises:

if the timeout event has occurred, storing an indication of a timeout
condition; and

wherein the timed program execution process performing a time-bounded
10 execution of the program comprises:

performing a) through e) if the timeout condition is not indicated.

6. The medium of claim 2, wherein the medium is further configured to
iteratively perform:

15 setting a timeout event;

the timed program execution process performing a time-bounded execution of the
program; and

disabling the timeout event.

20 7. The medium of claim 2, wherein the timeout process executes at a first
priority level, and wherein the medium is further configured to perform:

setting an execution priority level of the timed program execution process to a
second priority level, wherein the second priority level is below the first priority level.

25 8. The medium of claim 7,

wherein said performing a time-bounded execution of the program further
comprises:

storing an original execution priority level of the timed program execution process prior to said setting the execution priority level of the timed program execution process; and

wherein e) further comprises:

5 restoring the execution priority level of the timed program execution process to the original execution priority level.

9. The medium of claim 2,

wherein said initiating the timeout process comprises:

10 acquiring one or more resources for the time-bounded execution of the program; and

 initializing one or more resource managers for the one or more resources; and

wherein said terminating the timeout process comprises:

15 releasing the one or more resources for the time-bounded execution of the program; and

 un-initializing the one or more resource managers for the one or more resources.

20 10. The medium of claim 9,

wherein said initiating the timeout process further comprises:

 creating each of the one or more resource managers; and

wherein said terminating the timeout process further comprises:

 deleting each of the one or more resource managers.

25

11. The medium of claim 9,

wherein said initiating the timeout process further comprises:

 enabling each of the one or more resource managers; and

wherein said terminating the timeout process further comprises:

disabling each of the one or more resource managers.

12. The medium of claim 9, wherein said one or more resources comprises:
one or more memory pools for memory allocations during the time-bounded
5 execution of the program.

13. The medium of claim 9,
wherein, prior to a), the medium is further configured to perform:
storing a respective resource state for each of the one or more resources;
10 and
enabling each of the one or more resource managers.
wherein e) further comprises:
if the timeout event has occurred,
restoring the respective resource state for each of the one or more
15 resources; and
disabling each of the one or more resource managers.

14. The medium of claim 1, wherein said initiating the timeout process is
performed during said executing.
20

15. The medium of claim 1,
wherein the program comprises one or more sub-programs; and
wherein b) further comprises:
performing a) through e) for each of the one or more sub-programs.
25

16. The medium of claim 1, wherein said initiating the timeout process is
performed by the timed program execution process.

17. The medium of claim 1, wherein the rollback state comprises:

a rollback point, comprising an execution point in the program, and
an execution state of the timed program execution process at the rollback point.

18. The medium of claim 1, wherein the program comprises a subprogram,
5 and wherein said executing the program further comprises:
initiating another timeout process;
configuring another timeout event;
the timed program execution process performing a time-bounded execution of the
subprogram, comprising:
10 f) determining and storing another rollback state for the subprogram;
g) if the another timeout event has not occurred, executing the
subprogram, wherein, during said executing the subprogram, if the another timeout event
occurs,
h) the another timeout process setting the timed program execution
15 process to the another rollback state, and disabling the another timeout event; and
i) the timed program execution process resuming executing the
subprogram based on the another rollback state with another timeout condition;
j) performing a subprogram exit procedure;
disabling the another timeout event; and
20 terminating the another timeout process.

19. The medium of claim 18, wherein j) further comprises e).

20. The medium of claim 1, wherein the medium is further configured to
25 perform time-bounded execution of another program substantially in parallel with the
time-bounded execution of the program by:
initiating another timeout process;
configuring another timeout event;

the timed program execution process performing a time-bounded execution of the another program, comprising:

- f) determining and storing another rollback state for the another program;
- g) if the another timeout event has not occurred, executing the another
5 program, wherein, during said executing the another program, if the another timeout event occurs,
- h) the another timeout process setting the timed program execution process to the another rollback state, and disabling the another timeout event; and
- i) the timed program execution process resuming executing the
10 another program based on the another rollback state with another timeout condition;
- j) performing another program exit procedure;
disabling the another timeout event; and
terminating the another timeout process.

15 21. The medium of claim 1, wherein said executing the program further comprises:

- receiving a disable request from the program to disable the rollback state;
- disabling the rollback state in response to said disable request;
- receiving an enable request from the program to enable the rollback state;
- 20 enabling the rollback state in response to said enable request; and
- updating the rollback state for the program.

22. The medium of claim 1,
wherein the program comprises a machine vision application.

25 23. The medium of claim 1,
wherein the program comprises a numerical analysis application.

24. The medium of claim 1,

wherein the program comprises a text-based program.

25. The medium of claim 1, wherein the program comprises a graphical program.

5

26. The medium of claim 1, wherein the medium comprises a carrier medium.

27. The medium of claim 1, wherein the medium comprises a programmable hardware element.

10

28. A system for performing time-bounded execution of a program, wherein the system comprises:

a processor; and

a memory coupled to the processor, wherein the memory stores program instructions executable by the processor to:

initiate a timed program execution process, wherein the timed program execution process is operable to execute a program in a time-bounded manner;

initiate a timeout process;

set a timeout event;

wherein the timed program execution process is operable to perform a time-bounded execution of the program, wherein, in performing the time-bounded execution, the timed program execution process is operable to:

a) determine and store a rollback state for the program;

b) if the timeout event has not occurred, execute the program, wherein, during said executing, if the timeout event occurs,

c) the timeout process is operable to set the timed program execution process to the rollback state, and disable the timeout event; and

d) the timed program execution process is operable to resume executing the program based on the rollback state with a timeout condition; and

e) perform a program exit procedure;
disable the timeout event;
terminate the timeout process; and
terminate the timed program execution process.

5

29. A system for performing time-bounded execution of a program,
comprising:

means for initiating a timed program execution process, wherein the timed
10 program execution process is operable to execute a program in a time-bounded manner;

means for initiating a timeout process;

means for setting a timeout event;

means for the timed program execution process performing a time-bounded
execution of the program, comprising:

15 a) means for determining and storing a rollback state for the program;

b) if the timeout event has not occurred, means for executing the program,
wherein, during said executing, if the timeout event occurs,

c) means for the timeout process setting the timed program
execution process to the rollback state, and disabling the timeout event; and

20 d) means for the timed program execution process resuming
executing the program based on the rollback state with a timeout condition;

e) means for performing a program exit procedure;

means for disabling the timeout event;

means for terminating the timeout process; and

25 means for terminating the timed program execution process.

30. A medium configured to perform time-bounded execution of a program,
wherein the medium is configured to perform:

setting a timeout condition in a first process;
initiating execution of a program in a second process;
during said execution of the program, the first process determining if the timeout
condition has occurred;
5 if the timeout condition has occurred,
 the first process restoring the execution of the program to a pre-defined
program state; and
 the second process executing the program from the pre-defined state with
the timeout condition; and
10 the program exiting in the second process in response to the timeout
condition.